

A. REMARKS

1. General Remarks

The examiner rejected the claims under section 112 and 103. The claims have been amended to remove the section 112 rejections. In regard to the section 103 objections, applicant submits that the prior art, including the art cited by the examiner in the first office action, does not teach "applying batch submission to reduce network traffic when backing up and restoring directory limits in an operating system" (application, p. 2, lines 9-12). Therefore, the claims have been amended to more particularly point out the way in the invention reduces data transmission in the network (network traffic) when backing up and restoring directory limits in an operating system.

The need for applicant's invention was set forth as follows:

When backing up and restoring directory limits in an operating system, add, set and get routines work only on a single directory path. Since the operation on a single path may require multiple Add, Set or Get Application Program Interface (API) calls, restoration of a large amount of directory paths can generate a large amount of network traffic. Moreover, restoration of a large amount of directory paths can require sending and receiving the API and the data for each operation across the network. For example, in a network of computers, a first computer in the network may have a failed hard drive and a second computer may contain backup data for the first computer. An administrator may want to invoke a backup and restore routine. If there happened to be 64,000 target directories and all 64,000 directories had to be updated, there would be 128,000 round trip operations across the network. In some cases the backup and restore operations could take several days. What is needed is a way to reduce the time required to complete the backup and restore operations. In order to reduce the traffic, one way would be to batch all of the operations at one computer and send the batch to the second computer.

Moreover, the applicant addressed prior art batch submission and pointed out that the

prior art does not address the problem of applying batch submission to reduce network traffic when backing up and restoring directory limits in an operating system.

Batch processing has been used to increase the efficiency of data handling. United States Patent 5,889,896 discloses batch processing of raw image data which has been scanned into the memory of a processing system. System throughput is enhanced by permitting a workstation to execute a task on a batch of node index file records and to return the modified index file records to the stage of a work queue. However, the prior art does not address the problem of applying batch submission to reduce network traffic when backing up and restoring directory limits in an operating system.

Applicant's invention discloses a first computer connected to a second computer by a network. The first computer memory contains a plurality of directory limits needed for the restoration process of the operating system of a second computer in a back up file (application, pages 2 and 6). A first program, called a sender batch submission application interface (SBSA)¹ in the first computer groups the directory limits into a buffer and examines each record in the plurality of directory limits to determine whether a flag is not set or whether a path is not covered. If a flag is not set or a path is not covered, the record is invalid. Otherwise the record is valid (application, pages 2, 3, and 6). Valid records are placed in a batch buffer and invalid records are placed in a failed buffer. If all of the records pass, or if the percentage of failed records is less than a threshold, the batch buffer and count is sent to the second computer. (application, pages 3, 6, and 7) Upon receiving the batch and count, a second program² in the second computer, performs operations on each record in the batch buffer. The operations performed are to add a directory where one did not exist, set a directory limit if one existed, or to get information if required for an add operation or for a set operation. (application, pages 3 and 8) Because these operations are performed on the second

¹ Called a sender batch submission application interface (SBSA) in the application.

² Called a receiver batch submission application interface (RSBA) in the application.

computer, the network traffic that would take place if performed on the records in the first computer is reduced. In addition, application interface calls to process the records are issued locally at the second computer further increasing system efficiency. (application, pages 4, 8, and 9)

The examiner cited United States Patent 6,104,798 (the '798 patent). The '798 patent discloses a customer³ personal computer (the first computer) having a software application (client 110). The software application provides a graphical user interface and connects the personal computer to a carrier direct server (the second computer) (130) (5:62-66). Using the first computer's graphical user interface, customers enter add, change, or cancel orders (or import a batch of externally built orders). The software application applies edits to each order, notifies the customer of failed orders, and stores correct orders in a file to be uploaded to the server (5:67-6:7). The edits ensure that identifiers (called automatic number identifiers) are in the correct format (8:65-67). The orders that are in the correct format are sent in a batch file to the server (9:1-9). The server ensures that each new identifier is correct by comparing it to a database table (9:12-14). A count⁴ is automatically updated to reflect the correct number of records in the batch (15:43-44).

Therefore, the '798 patent discloses storing correct orders in a batch file to send to a server where each new identifier is compared to a database table. The data that is sent is not needed by the server computer for its own operation, and no actual operations are performed on the data. In other words, the comparison is merely a check and not a modification. In contrast, applicant's invention applies batch processing to directory limits necessary for the restoration of the second computer's operating system, and sends valid directory limits to the second computer where the second computer performs actual

³ The customer is a Reseller Enhanced Local Exchange Carrier Interface or a Carrier Network Service. (8:58-62).

⁴ The '798 patent's count is called a Batch Trailer Record.

operations on the directory limits.

2. Claim Rejections – 35 USC § 103(a)

The Examiner rejected claims 1, 3, 5, 6, 8, 12 14 and 16 under 35 USC 103(a) as being unpatentable over Lickiss et al (U.S. Patent No. 6,104,798). The examiner rejected claim 4 under 35 U.S.C. 103(a) as being unpatentable over Lickiss et al (U.S. Patent No. 6,104,798) in view of Martinez et al (United States Patent 6,119,229). The examiner rejected claims 13 and 15 as being unpatentable over Lickiss et al (U.S. Patent No. 6,104,798) in view of Forecast (United States Patent 6,230,200).

MPEP § 2143.01 states:

The mere fact that references can be combined or modified does not render the resultant combination obvious unless the prior art also suggests the desirability of the combination. *In re Mills*, 916 F.2d 680, 16 USPQ2d (Fed. Cir. 1990).

The Federal Circuit has several times expressly addressed the issue of how to evaluate an alleged case of *prima facie* obviousness to determine whether it has been properly made. Thus, *In re Geiger*, *supra*, stated, in holding that the PTO “failed to establish a *prima facie* case of obviousness”:

Obviousness cannot be established by combining the teachings of the prior art to produce the claimed invention, absent some teaching, suggestion, or incentive supporting the combination. *ACS Hospital Systems, Inc. v. Montefiore Hospital*, 732 F.2d 1572, 1577, 221 USPQ 929, 933 (Fed. Cir. 1984).

As discussed in General Remarks above, the '798 patent does not teach or disclose batch processing of directory limits necessary for the restoration of a second computer's

operating system where the second computer performs actual operations on the directory limits. Based on amended claim limitations, applicant asserts that the Lickiss patent is no longer a proper reference for rejecting the claims because it does not teach the use of batch operations for processing of directory limits in a back up file of a first computer that are necessary for restoration of a second computer's operating system where the second computer performs operation on the directory limits. Furthermore, neither Lickiss, Martinez nor Forecast disclose using a threshold to determine whether to determine whether or not to send a batch.

B. CLAIM AMENDMENTS

What is claimed.

1. (Currently amended) A programmable apparatus comprising:
 - a first computer having a first computer memory containing a plurality of directory limits for the operating system of a second computer in a back up file;
 - a SBSAsender batch submission application interface in the first computer memory ;
a network connecting the first computer and the second computer;
 - athe second computer having a second computer memory;
 - a RBSA-receiver batch submission application interface in the second computer memory that only accepts a buffer and count;
 - the first computer being directed by the SBSA-sender batch submission application interface to examine each record in the plurality of directory limits to determine a backup file and determine whether each record is a valid record or an invalid recordwhether a flag is not set or whether a path is not covered;
 - identifying a record where a flag is not set or a path is not covered as an invalid record and any other record as valid record;
 - to place the valid records in a batch buffer and the invalid records in a failed buffer ;
 - to determine the number of records in the batch buffer -and transmitting to transmit atthe BCbuffer and count to the second computer; and
 - the second computer, being directed by the receiver batch submission application interfaceRBSA, receives the BCbuffer and count and performs an operation on each record in the BCbuffer and count to add a directory, set a directory limit, or to get information if required to add a directory or to set a directory limit;
 - whereby transmission of data over the network when backing up and restoring directory limits in the operating system is reduced.

2. (Currently amended) The programmable apparatus of claim 1 wherein said sender batch submission application interface^{SBSA} directs said first computer to determine X where X is the total count of records in the file, to determine Y, where Y is the total number of invalid records, and to determine a number by dividing the total number of invalid records by the by the total count of records in the file, determine Y/X, and where Y/X the number is less than or equal to Z, where Z is a predetermined number, then said sender batch submission application interface^{SBSA} directs said first computer to send the batch and count^{BC}; and where the number Y/X is greater than the predetermined number Z, where Z is a predetermined number, said sender batch submission application interface^{SBSA} directs said first computer to display a report that an error threshold has been exceeded.

3. (Currently amended) The programmable apparatus of claim 1 wherein said RBSAreceiver batch submission interface directs said second computer to perform an operation on each record, and where an operation on a record failed, to generate an error record.

4. (Currently amended) The programmable apparatus of claim 1 wherein said RBSAreceiver batch submission interface directs said second computer to determine whether any records failed to be operated on and, responsive to a determination that there were records that failed to be operated on, to return the records that failed to be operated on to the first computer.

5. (Currently amended) The batch and count of claim 1 wherein the count of records is equal to the total count of records in the backup file minus the number of invalid records.

6. (Currently amended) A computer readable memory for causing a computer, having a file containing a plurality of records, to validate the plurality of records for transmission to a second computer comprising:

a computer readable storage medium;

an SBSAsender batch submission application interface stored in said storage medium; the storage medium so configured by said SBSAsender batch submission application interface, causes the computer to examine each record in a plurality of directory limits and to determine whether each record is a valid record or an invalid record;

to place the valid records in a batch buffer and the invalid records in a failed buffer;

to determine the number of records in the batch buffer and to transmit a BC batch and count to the second computer;

wherein a record is invalid if a flag is not set or a path is not covered, and any other record is valid; and

wherein the batch and count is transmitted to a second computer so that a receiver batch submission application interface in a memory of the second computer can add a directory, set a directory limit or and get information in order to add a directory or to set a directory limit:

whereby transmission of data over the network when backing up and restoring directory limits in the operating system is reduced.

7. (Currently amended) The SBSAsender batch submission application interface of claim 6, wherein said SBSA directs said computer to determine X, where X is the total count of records in the file, to determine Y, where Y is the total number of invalid records, and to determine Y/X, a number by dividing the total number of invalid records by the total count of records in the file, and where Y/X the number is less than or equal to Z, where Z is a

predetermined number, then said SBSAsender batch submission application interface directs said computer to send the BCbatch and count; and where Y/Xthe number is greater than Z, where Z is a the predetermined number, said SBSAsender batch submission application interface directs said computer to display a report that an error threshold has been exceeded.

8. (Currently amended) The BCbatch and count of claim 6 wherein the count of records is equal to the total number of records in the file containing a plurality of records minus the number of invalid records.

9. (Currently amended) A computer implemented process to accomplish pre-submission validation and batch submission of directory limits from a backup file comprising: using a SBSAsender batch submission application interface in the memory of a first computer, performing the following steps;

initializing Xa total count of records in a file and Ythe total number of invalid records in the file;

retrieving a record;

setting X to X + 1adding one to the total count of records in the file;

examining the record to determine whether a flag is not set or a path is not covered;

identifying a record where a flag is not set or a path is not covered as an invalid record and any other record as a valid record;

responsive to determining that there is a validation error in the record, setting Y = Y + 1adding one to the total number of invalid records in the file and storing the invalid record in a failed buffer;

responsive to determining that there is no validation error in the record; storing the

record in a batch buffer ;
determining whether the last record has been retrieved;
responsive to a determination that the last record has been retrieved, determining whether the total number of invalid records in the file \geq is greater than θ_{zero} ;
responsive to a determination that \geq the total number of invalid records in the file is greater than θ_{zero} , sending a BCbatch and count;
using a RBSA receiver batch submission application interface in the memory of a second computer, performing the following steps:
receiving the BCbatch and count;
retrieving a record;
determining whether the last record has been retrieved;
operating on the record to add a directory, set a directory limit or get information in order to add a directory or to set a directory limit and determining whether the operation failed; and
responsive to a determination that the operation failed, generating an error record;
whereby transmission of data over the network when backing up and restoring directory limits in the operating system is reduced.

10. (Currently amended) The computer implemented process of claim 9 -wherein, using the RBSA receiver batch submission application interface in the memory of the second computer, performing the follow steps:

responsive to determining that the last record has been retrieved, determining whether there have been any failures; and
responsive to a determination that there have been failures, returning the error

records.

11. (Currently amended) The computer implemented process of claim 9 ~~BC~~ of claim 9 wherein the count of records is equal to the total number of records in the backup file minus the number of invalid records.

12. (Currently amended) A method for reducing remote APIapplication interface network traffic and increasing APIapplication interface performance in a network having a first computer and a second computer comprising:

validating a plurality of records in a back up file, comprising a plurality of directory limits for the operating system of a second computer, prior to submission in the first computer by examining each of the plurality of records to determine whether a flag is not set or a path is not covered;

identifying a record where a flag is not set or a path is not covered as an invalid record and any other record as a valid record;

storing said records in a batch and count~~BC~~;

transmitting said batch and count~~BC~~ to said second computer;

receiving said ~~BC~~batch and count in said second computer; and

operating on said records in said second computer to add a directory, set a directory limit or get information in order to add a directory or to set a directory limit;

whereby transmission of data over the network when backing up and restoring directory limits in the operating system is reduced.

13. (Currently amended) The validating records prior to submission of claim 12 further comprising the steps of:

initializing Xa total count of records in a file and Ythe total number of invalid records in

the file;

retrieving a record;

~~setting X to X + 1~~ adding one to the total count of records in the file; and
responsive to determining that there is a validation error, ~~setting Y = Y + 1~~ adding
one to the total number of invalid records in the file.

14. (Currently amended) The method of claim 12 wherein the storing step of claim 12
further comprising further comprises the steps of:

storing an invalid record in a failed buffer and storing a valid record in a batch
buffer; and

calculating a count of records equal to the count of total records read in the
validating records prior to submission step minus the number of invalid records.

15. (Currently amended) The method of claim 12 wherein the sending step of claim 12
further comprising comprises the steps of:

responsive to a determination that the last record has been retrieved, determining
whether ~~Y~~ the total number of invalid records in the file is greater than
0zero;

responsive to a determination that ~~Y~~ the total number of invalid records in the file
is greater than 0zero, transmitting the batch and count.

16. (Currently amended) The method of claim 12 wherein the operating step of claim 12
further comprising comprises the steps of:

operating on the record and determining whether the operation failed; and
responsive to a determination that the operation failed, generating an error record.

For the foregoing reasons, the Applicant submits that the claims of the present application are not fairly taught by any of the references of record, taken either alone or in combination. Therefore, allowance of the present application is in order, and is requested.

Respectfully submitted,

Rudolf O. Siegesmund

Rudolf O. Siegesmund
Registration No. 37,720
Suite 2000
4627 N. Central Expressway
Dallas, Texas 75205-4017
214-528-2407
FAX 214-528-2434
Attorney for Applicant

Certificate of Transmission

I hereby certify that this correspondence is being facsimile transmitted to the Patent and Trademark Office Fax No. (703) 872-9306.

On 6/29/04
Date

Rudolf O. Siegesmund
Rudolf O. Siegesmund